Class name : SY CSE(IOT)
Rollno : 2007
Urn no : 1022101007
Batch : S1

EXPERIMENT 2

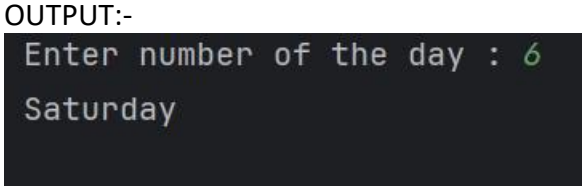
2. Implement python programs to demonstrate decision control and looping
Statements

a) Write a python Program to read a number and display corresponding day using
if_elif_else?

CODE:-

```
#Write a python Program to read a number and display corresponding day using
if_elif_else?
a = int(input("Enter number of the day : "))
if a == 1:
   print("Monday")
elif a==2:
   print("Tuesday")
elif a==3:
   print("Wednesday")
elif a==4:
   print("Thursday")
elif a==5:
   print("Friday")
elif a==6:
   print("Saturday")
elif a==7:
   print("Sunday")
else:
   print("Invalid Input")
```

OUTPUT:-



b) Write a program to create a menu with the following options
1. TO PERFORM ADDITITON
2. TO PERFORM SUBTRACTION
3. TO PERFORM MULTIPICATION
4. TO PERFORM DIVISION
Accepts users input and perform the operation accordingly

CODE:-

```
while(True):
   print("1. TO PERFORM ADDITITON")
   print("2. TO PERFORM SUBTRACTION")
   print("3. TO PERFORM MULTIPICATION")
   print("4. TO PERFORM DIVISION")
   print("5. Exit")
   a = int(input("Choice any one of the above option : "))
   if a == 1:
      b = int(input("Enter first number : "))
      c = int(input("Enter second number : "))
      print("The addition of 2 number is ",b+c)
   elif a == 2:
      b = int(input("Enter first number : "))
      c = int(input("Enter second number : "))
      print("The Subtraction of 2 number is ",b+c)
   elif a == 3:
      b = int(input("Enter first number : "))
      c = int(input("Enter second number : "))
      print("The Multiplication of 2 number is ",b+c)
   elif a == 4:
      b = int(input("Enter first number : "))
      c = int(input("Enter second number : "))
      print("The Division of 2 number is ",a/b)
   elif a == 5:
      print("!!!!!!!Exiting the program!!!!!!!")
      break
   else:
      print("Invalid input ")
```

OUTPUT:-

```
C:\Users\Dragon\AppData\Local\Programs\Python\Python38\python.exe D:\Python\Experiments\Exp2\Exp2b.py
1. TO PERFORM ADDITITON
2. TO PERFORM SUBTRACTION
3. TO PERFORM MULTIPICATION
4. TO PERFORM DIVISION
5. Exit
Choice any one of the above option : 2
Enter first number : 10
Enter second number : 20
The Subtraction of 2 number is  30
1. TO PERFORM ADDITITON
2. TO PERFORM SUBTRACTION
3. TO PERFORM MULTIPICATION
4. TO PERFORM DIVISION
5. Exit
Choice any one of the above option : 5
!!!!!!!Exiting the program!!!!!!!

Process finished with exit code 0
```

c) To generate prime numbers within an interval using for and while statements.

CODE:-

```
def is_prime(num):
    """Check if a number is prime."""
    if num <= 1:
        return False
    if num <= 3:
        return True
    if num % 2 == 0 or num % 3 == 0:
        return False
    i = 5
    while i * i <= num:
        if num % i == 0 or num % (i + 2) == 0:
            return False
        i += 6
    return True

def generate_primes(lower, upper):
    """Generate prime numbers in the given interval [lower, upper]."""
    prime_numbers = []
    for num in range(lower, upper + 1):
        if is_prime(num):
            prime_numbers.append(num)
```

```
    return prime_numbers

# Get user input for the interval
lower_bound = int(input("Enter the lower bound of the interval: "))
upper_bound = int(input("Enter the upper bound of the interval: "))

# Generate and print prime numbers in the interval
primes = generate_primes(lower_bound, upper_bound)
print(f"Prime numbers between {lower_bound} and {upper_bound} are: {primes}")
```

OUTPUT:-

```
Enter the lower bound of the interval: 1
Enter the upper bound of the interval: 60
Prime numbers between 1 and 60 are: [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59]

Process finished with exit code 0
```

d) Write a python program to construct the following pattern, using a nested for loop

CODE:-

```
def Diamond(rows):
    n = 1
    for i in range(1, rows + 1):
        # loop to print spaces
        for j in range(1, (rows - i) + 1):
            print(end=" ")

        # loop to print star
        while n != (i + 1):
            print("*", end=" ")
            n = n + 1
        n = 1

        # line break
```

```
        print()

    k = 0
    n = 0
    for i in range(1, rows + 1):
        # loop to print spaces
        for j in range(1, k + 1):
            print(end=" ")
        k = k + 1

        # loop to print star
        while n <= (rows - i):
            print("*", end=" ")
            n = n + 1
        n = 0
        print()

    # Driver Code


# number of rows input
rows = 5
Diamond(rows)
```

OUTPUT:-